This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently Amended) A computer system having computer instruction modules for batching tasks in a build process, the modules comprising:

a project file comprising at least one task, the at least one task for consuming a plurality of items, each of the plurality of items associated with a value representing a set of attributes; and

a build engine that receives [[a]] the project file and generates executable code from the project file, ~~the project file comprising at least one task, the at least one task consuming at least one item of a plurality of items, each of the plurality of items associated with a value representing a set of attributes,~~ the build engine ~~batching~~ grouping into buckets the plurality of items of the task of the project file based on the value representing the set of ~~attibutes~~ attributes for each item, each bucket representing a particular value and containing at least one of a plurality of the items of the task having the represented value, and the build engine executing the task once for each bucket thereof.

2. (Original) The system of claim 1, wherein the value representing the set of attributes comprises an identity of the item.

3. (Original) The system of claim 1, wherein the set of attributes comprises at least one attribute of a plurality of attributes, the at least one attribute associated with a scalar.

4. (Original) The system of claim 2, wherein the build engine batches the plurality of items based on the identity associated with each of the plurality of items, generating a separate batch for each unequal identity.

5. (Original) The system of claim 4, wherein the at least one task is executed once for each generated batch.

6. (Currently Amended) A method for batching build tasks comprising:

receiving a project file comprising at least one [[a]] build task, the at least one build task consuming task input, the task input comprising ~~at least one item of~~ a plurality of items, each of the plurality of items associated with an identity comprising a value for a set of attributes;

analyzing the task input to generate at least a first bucket for items of the build task associated with a first identity and a second bucket for items of the build task associated with a second identity such that each of the first and second buckets represents a particular identity and contains at least one of a plurality of the items of the build task having the represented identity, wherein the first identity and the second identity are unequal; and

executing the build task once for the first ~~each~~ generated bucket and once for the second generated bucket.

7. (Original) The method of claim 6, wherein a plurality of satellite assemblies are automatically generated.

8. (Original) The method of claim 6, wherein a plurality of assemblies are automatically generated.

9. (Original) The method of claim 6, wherein an executable is automatically invoked once for each entry in a list.

10. (Original) The method of claim 6, wherein the plurality of items is filtered based on an attribute.

11. (Original) The method of claim 6, wherein the plurality of items is filtered based on a plurality of attributes.

12. (Currently Amended) The method of claim 6, wherein only out-of-date batches are processed as determined based on timestamp checking of each item.

13. (Original) The method of claim 6, wherein the at least one item is of a first type.

14. (Original) The method of claim 13, wherein executing the build task transforms the at least one item of the first type to an item of a second type.

15. (Original) The method of claim 6, wherein the identity comprises at least one scalar of a plurality of scalars, the at least one scalar comprising a value for one attribute.

16. (Original) The method of claim 15, wherein the attribute value may be pre-defined.

17. (Original) The method of claim 16, wherein the attribute value may be declared.

18. (Currently Amended) A method for batching build tasks comprising:

receiving a project file comprising at least one [[a]] build task, the at least one build task consuming a plurality of items, each of the plurality of items associated with an identity comprising a value for a set of attributes;

grouping items of the build task having a first particular identity into a first bucket such that the first bucket represents the first particular identity and contains at least one of a plurality of the items of the build task having the represented first identity;

grouping items of the build task having a second particular identity into a second bucket such that the second bucket represents the second particular identity and contains at least one of a plurality of the items of the build task having the represented second identity, wherein the first identity and the second identity are unequal; and

executing the build task once for the first bucket and once for the second bucket.

19. (Original) The method of claim 18, further comprising automatically generating a plurality of assemblies.

20. (Original) The method of claim 18, further comprising automatically invoking an executable once per entry in a list.

21. (Original) The method of claim 18, further comprising filtering the plurality of items based on an attribute.

22. (Original) The method of claim 18, further comprising filtering the plurality of items based on a plurality of attributes.

23. (Currently Amended) The method of claim 18, further comprising processing only out-of-date batches as determined based on timestamp checking of each item.

24. (Original) The method of claim 18, wherein at least one item of the plurality of items is of a first type.

25. (Original) The method of claim 24, wherein executing the build task transforms the at least one item of the first type to an item of a second type.

26. (Currently Amended) A computer-readable medium comprising computer-executable instructions for:

receiving a project file comprising at least one [[a]] build task, the at least one build task consuming a plurality of items, each of the plurality of items associated with an identity comprising a value for a set of attributes;

grouping items of the build task having a first particular identity into a first bucket such that the first bucket represents the first particular identity and contains at least one of a plurality of the items of the build task having the represented first identity;

grouping items of the build task having a second particular identity into a second bucket such that the second bucket represents the second particular identity and contains at least one of a plurality of the items of the build task having the represented second identity, wherein the first identity and the second identity are unequal; and

executing the build task once for the first bucket and once for the second bucket.

27. (Original) The computer-readable medium of claim 26, wherein executing the build task programmatically generates a plurality of assemblies.

28. (Original) The computer-readable medium of claim 26, wherein executing the build task programmatically invokes an executable once per entry in a list.

29. (Original) The computer-readable medium of claim 26, wherein executing the build task filters the plurality of items.

30. (Original) The computer-readable medium of claim 26, wherein executing the build task filters the plurality of items based on a plurality of attributes.